Recognizing emotions by analyzing facial expressions

Aruna Shenoy
a.1.shenoy@herts.ac.uk

Tim MGale
t.gale@herts.ac.uk

Ray Frank
r.j.frank@herts.ac.uk

Neil Davey
n.davey@herts.ac.uk

School of Computer Science
University of Hertfordshire
UK

Abstract

Recognizing expressions is a key part of human social interaction, and processing of facial expression information is largely automatic. However, it is a non-trivial task for a computational system. Our purpose of this work is to develop Computational models capable of differentiating between ranges of human Facial expressions. The Gabor feature is effective for facial image representation. The Gabor feature dimensionality is so high that a dimensionality reduction technique such as PCA must be applied. Classification of various classes of expressions can be achieved by training and then testing with a Support Vector Machine (SVM).

1 Introduction

According to Ekman and Friesen (Ekman et al., 1971) there are six easily discernible facial expressions: anger, happiness, fear, surprise, disgust and sadness. Moreover these are readily and consistently recognized across different cultures (Batty et al., 2003). In the work reported here we show how a computational model can identify facial expressions from simple facial images. In particular we show how smiling faces and neutral faces can be differentiated.

We first pre-process the images using Gabor Filters (Jain et al., 1991; Movellan 2002). The features of the face (or any object for that matter) can be aligned at any angle. Using a suitable Gabor filter at the required orientation, certain features can be given high importance and other features less importance. Usually, a bank of such filters is used with different parameters and later the resultant image is a *L2 max* (at every pixel the maximum of feature vector obtained from the filter bank) superposition or average of the outputs from the filter bank. Gabor filters are interesting because simple cells in the visual cortex are known to be selective for the following four parameters: the *x*, *y* location in visual space, the preferred orientation, and the preferred spatial frequency (Daugman, 1985).

Recent work on these suggests that the various 2D receptive field profiles encountered in populations of simple cells are well described by the family of 2D Gabor filters (Daugman, 1985).

Data presentation plays an important role in any type of recognition. High dimensional data, such as the output of the Gabor filters of the face images, must be reduced to a manageable low dimensional data set by using a technique such as Principal Component Analysis (PCA). The Intrinsic Dimension (ID) (Grassberger et al., 1983), which is the true dimension of the data, is often much less than the original dimension of the data.

2 Background

We begin with a simple experiment to classify two expressions: neutral and smiling. The image is pre-processed by using a bank of Gabor filters. A Support Vector Machine (SVM) (Chih-Chung et al., 2001) based classification technique is used.

2.1 Gabor Filters

A Gabor filter can be applied to images to extract features aligned at particular angles (orientations). Gabor filters possess the optimal localization properties in both spatial and frequency domains, and they have been successfully used in many applications (Zheng et al., 2004a). A Gabor filter is a function obtained by modulating a sinusoidal with a gaussian function. The useful parameters of a Gabor filter are orientation and frequency. It is used to enhance certain features that share an orientation and/or frequency and thereby enables useful pre-processing required for facial expressions, recognition and analysis to be carried out. The Gabor filter is thought to mimic the simple cells in the visual cortex. The various 2D receptive-field profiles encountered in populations of simple cells in the visual cortex are well described by an optimal family of 2D filters (Daugman, 1985). In our case a Gabor filter bank is implemented on face images with 8 different orientations and 5 different frequencies.

A Gabor filter can be one or two dimensional (2D). A 2D Gabor filter is expressed as a Gaussian modulated sinusoid in the spatial domain and as shifted Gaussian in the frequency domain. Recent studies on modeling of visual cortical cells (Kulikowski, 1982) suggest a tuned band pass filter bank structure. These filters are found to have Gaussian transfer functions in the frequency domain. Thus, taking the Inverse Fourier Transform of this transfer function gives characteristics closely resembling Gabor filters.

A well designed Gabor filter bank can capture the relevant frequency spectrum in all directions. Phase can be taken as a feature because it contains information about the edge locations and other such details in the image; amplitude at every pixel can be taken as a feature as it contains some oriented frequency spectrum at every point of the image. We can extract many meaningful features using the Gabor filter family. Experimental results in texture analysis and character analysis demonstrate these features in the capture of local information with the different frequencies and orientations in the image (Zheng et al., 2004a).

The Gabor filter is a Gaussian (with variances S_x and S_y along x and y-axes respectively) modulated by a complex sinusoid (with centre frequencies U and V along x and y-axes respectively) described by the following equation:-

$$g(x,y)=\frac{1}{2\pi S_x S_y}\exp\left[-\frac{1}{2}\left\{\left(\frac{x}{S_x}\right)^2+\left(\frac{y}{S_y}\right)^2\right\}+2\pi j(Ux+Vy)\right] \tag{1}$$

The variance terms S_x and S_y dictate the spread of the band pass filter centered at the frequencies U and V in the frequency domain. This filter is complex and the plot of the Real and Imaginary parts of $g(x,y)$ is shown in Figure 1 :-

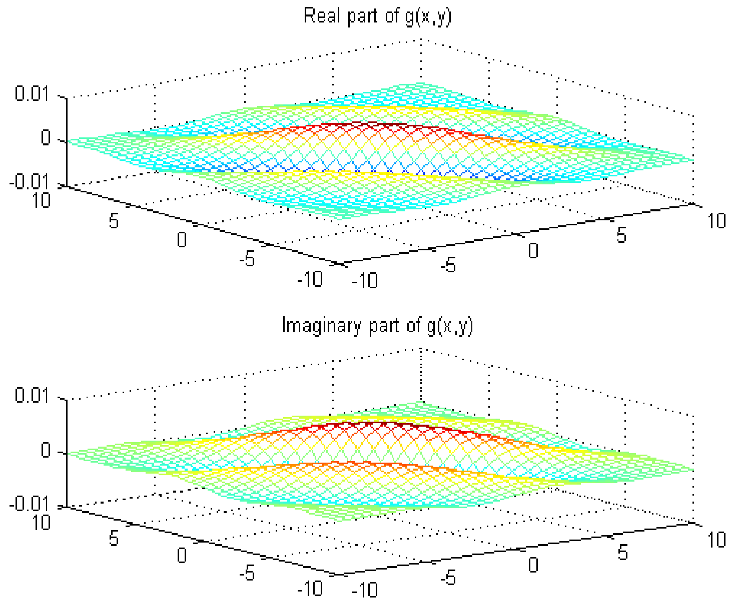


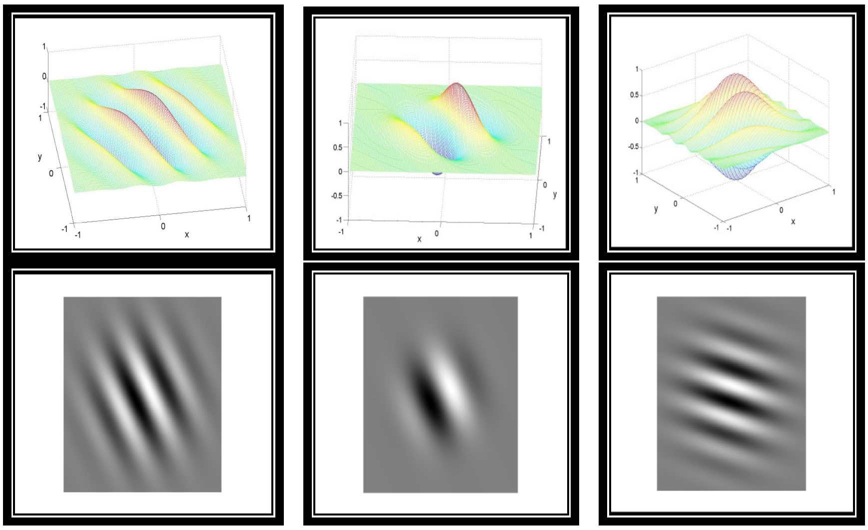
Figure 1: Plot of Real and Imaginary part of Gabor filter

It is found that as the 2D Gabor filter is applied to the images, the edges are smoothened out in all directions due to the presence of the Gaussian term. Each filter can be designed to pick out particular image features in orientation and the required frequency.

A Gabor filter can be best described by the following parameters:

1. The S_x and S_y of the gaussian explain the shape of the base (circle or ellipse).
2. The frequency (*f*) of the sinusoid.
3. The orientation (Θ) of the applied sinusoid.

Figures 2 and Figure 3 show examples of various Gabor filters



(a) (b) (c)

Figure 2: Figures (a), (b), (c) are examples of Gabor filter with different frequencies and orientations. Top row shows their 3D plots and the bottom row, the intensity plots of their amplitude along the image plane.

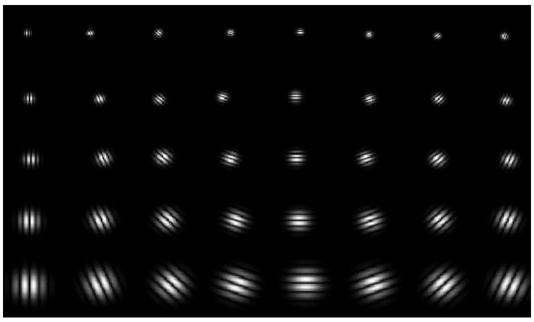


Figure 3: Gabor filters: Real part of the Gabor kernels at five scales and eight orientations

Figure 5 and Figure 6 show the effect of applying a particular Gabor filter on Figure 4 which is an image with lines at various angles. The highlighted lines in Figure 5 and Figure 6 shows the way the Gabor filter exaggerates lines at particular orientations.

Figure 4: Image with lines at various angles

Figure 5: Frequency, $f = 12.5$ and orientation, $\theta = 135$ degrees

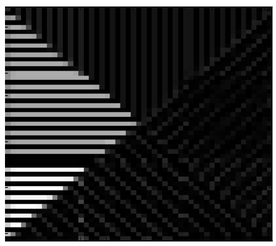
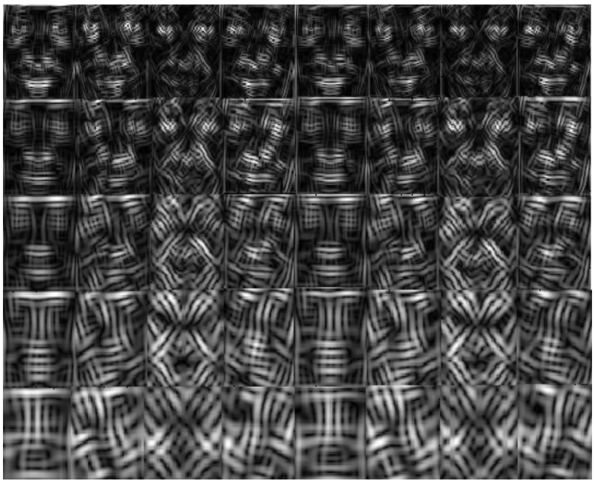


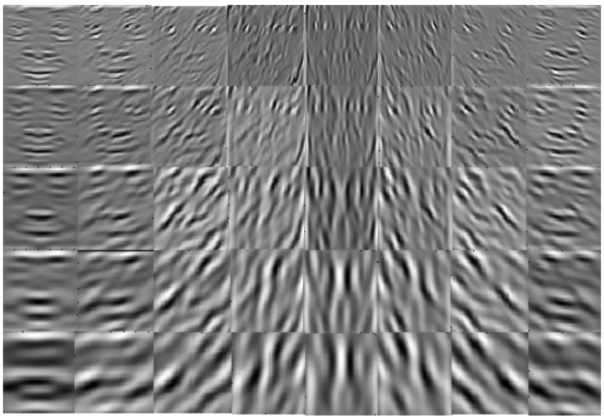
Figure 6: Frequency, f=25 and orientation, θ=0 degrees

Figure 8 shows the effect of applying variety of Gabor filters to the image shown in Figure 7. Note how the features at particular orientations are exaggerated.

Figure 7: Sample Image of size 64 × 64



(a)



(b)

Figure 8: Convolution outputs of a sample image shown in Figure 7 and the Gabor kernels (Fig. 3). (a) Magnitude part of the convolution outputs. (b) Real part of the convolution outputs.

Analytical methods make use of Gabor jets at specific points on the face which are vital feature points (fiducial points). There are different methods for identifying or locating these feature points. For elastic graph based analytic methods, a graph is first placed at an initial location and deformed using jets to optimize its similarity with a model graph. Non-graph based methods locate feature points manually or by color or edge etc. Once the location process is completed, recognition can then be performed using Gabor jets extracted from those feature points (Shen 2004).

Holistic methods on the other hand normally extract features from the whole face image. An augmented Gabor feature vector is thus created which produces a very large data for the image. Every pixel is then represented by a vector of size 40 and demands dimensionality reduction before further processing. So a 64 × 64 image is transformed to size 64 × 64 × 5 × 8. So, the feature vector consists of all useful information extracted from different frequencies, orientations and from all locations, and hence is very useful for expression recognition. Once the feature vector is obtained, it can be handled in various ways. We have performed the following operations and any one of them can be used for the feature extraction:

a) The final image can be of the average of the magnitudes of the Gabor filter coefficients at each location in the filter bank output.

b) The pixel value in the final image would be the *L2 max* norm value of the feature vector obtained from the Gabor filter bank

The *L2 max* norm Superposition principle is used on the outputs of the filter bank and the figure 10 shows the output for the original image of figure 9. Similarly the outputs of the 40 filter banks can also be averaged or summed to give an output as in figure 11 shown below.

Figure 9: Original Image used for the Filter bank

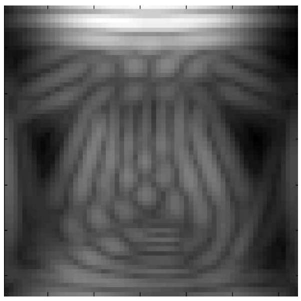


Figure 10: Superposition output (*L2 max* norm)

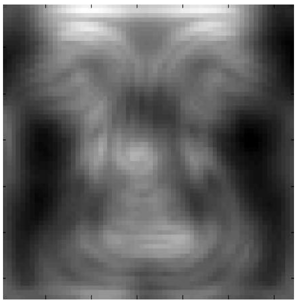


Figure 11: Average Output

2.2 Classification using Support Vector Machines

A number of classifiers can be used in the final stage for classification. We have concentrated on the Support Vector Machine. Support Vector Machines **(SVM)** are a set of related supervised learning methods used for classification and regression. They belong to a family of generalized linear classifiers. The SVM finds the optimal separating hyper-plane that has the maximal margin of separation between the classes, while having minimum classification errors. This means the SVM classifier tries to find the plane which separates the two different classes such that it is equidistant from the members of either class which are nearest to the plane. SVM's are used extensively for a lot of classification tasks such as: handwritten digit recognition (Cortes et al., 1995) or Object Recognition (Blanz et al., 1996). SVM's can be slow in test phase, although they have a good generalization performance. In total the SVM theory says that the best generalization performance can be achieved with the right balance between the accuracy attained on the training data and the ability to learn any training set without errors, for the given amount of training data. The SVM shows better classification accuracy than Neural Networks (NNs) if the data set is small. Also, the time taken for training and predicting the test data is much smaller for a SVM system than for a NN (Zheng et al., 2004b).

In short, they can be explained as a classifier which finds the optimum plane that performs the classification task by constructing a hyperplane in a multidimensional space that separate cases of different class labels.

In this example, the objects belong either to class GREEN or RED. The separating line defines a boundary on the right side of which all objects are GREEN and to the left of which all objects are RED. Any new object falling to the right is labeled, i.e., classified, as GREEN or classified as RED if it falls to the left of the separating line.

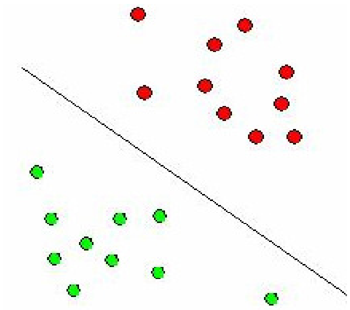


Figure 12: A Linear Classifier

Most classifications are not this simple, and a more complicated example is shown in Figure 13. In this example, it needs a curve rather than a straight line to separate the two classes.

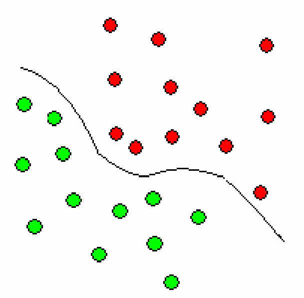


Figure 13: A non Linear Classifier.

A SVM rearranges the original objects (data points) according to a mathematical function (kernels) and transforms it into a feature space which allows the classification to be accomplished more easily, and is illustrated in Figure 14.

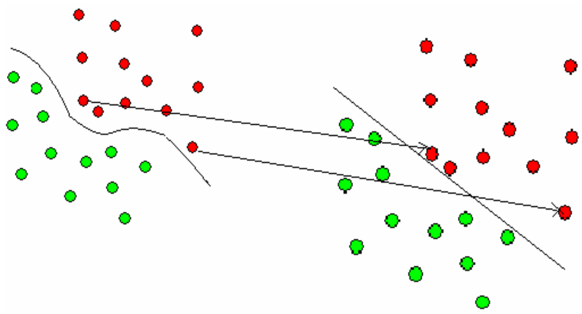


Figure 14: Transformation from input space to Feature space by the Support Vector Machine

We have used the LIBSVM tool (Chih-Chih, 2001) for SVM classification.

2.3 Principal Component Analysis

Principal Component Analysis (PCA) transforms higher dimensional datasets into lower dimensional uncorrelated outputs by capturing linear correlations among the data, and preserving as much information as possible in the data. PCA transforms data from the original coordinate system to the principal axes coordinate system such that the principal axis passes through the maximum possible variance in the data. The second principal axis passes through the next largest possible variance and this is orthogonal to the first axis. This is repeated for the next largest possible variances and so on. All these axes are orthogonal to each other. On performing the PCA on the high dimensional data, eigenvalues or principal components are thus obtained (Smith, 2002). The required dimensionality reduction is obtained by retaining only the first few principal components.

The PCA is used to project a D-dimensional dataset X onto an uncorrelated d- dimensional dataset Y, where $d \leq D$, by capturing the linear correlation between the data and preserving as much information as possible. In other words, the aim is to find a set of d orthogonal vectors in the data space that account for as much as possible of the variance of the data. Projecting the data from their original D-dimensional space onto the d-dimensional subspace spanned by these vectors then performs a dimensionality reduction that often retains most of the intrinsic information in the data. The variances measured on these orthogonal axes are the eigenvalues of the Principal components (Smith, 2002).

The Principal Components have the following properties: They can be ranked by decreasing order of "importance". The first few most "important" Principal Components account for most of the information in the data. In other words, one may then discard the original data set, and replace it with a new data set with the same observations, but fewer variables, without throwing away too much information.

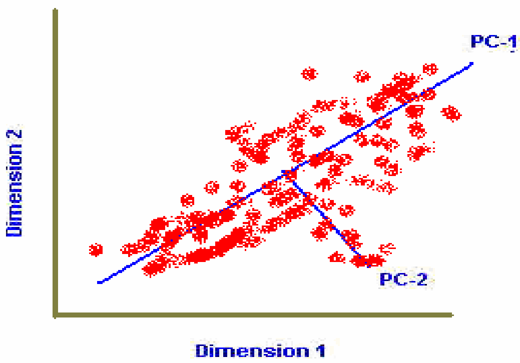


Figure 15**:** Figure shows the first two consecutive principal components.

The principal components are:
1. Orthogonal (at right angles) to each other.
2. They are uncorrelated.

3 Experiments and Results

We experimented on 120 faces (60 male and 60 female) each with two classes, namely, Neutral and smiling (60 faces for each expression). The images are from The FERET dataset (Philips et al., 2003).

Figure 16: Example FERET images used in our experiments and then cropped to the size of 128 × 128 to extract the facial region.

The training set was 80 faces (with 40 female, 40 male and equal numbers of them with neutral and smiling). Two test sets were created. In both test sets the number of each type of face is balanced. For example, there were 5 smiling male faces, 5 smiling female faces and 5 neutral male faces. For the purpose of comparison, the SVM classification was performed on the raw face images (150 × 130). With all faces aligned based on their eye location, a 128 × 128 image was cropped from the original (150 × 130). The resolution of these faces is reduced to 64 × 64. The classification was then performed on these images after reducing the dimensionality of the images by PCA. Later classification was performed on the Gabor pre processed image. The results are shown in Table 1. PCA was used to reduce dimensionality of the image of size 64 $\times$64 dimensions (4096 pixels) to 350 i.e. taking the first 350 PCA components. The first set has images which are easily discernible smiling faces. The second test set has smiling and neutral faces, but the smiling faces are not easily discernible.

Figures 17 and 18 show the first 5 eigenfaces from the neutral faces (top row), smiling faces (bottom row) and the complete training set.

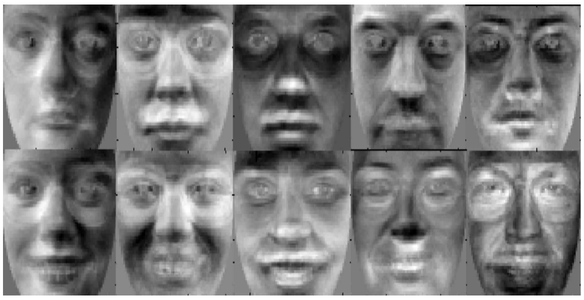


Figure 17: The top row shows the first 5 Eigen faces of all the neutral faces of the data set. The bottom row shows the first 5 Eigen faces of the smiling faces of the data set.

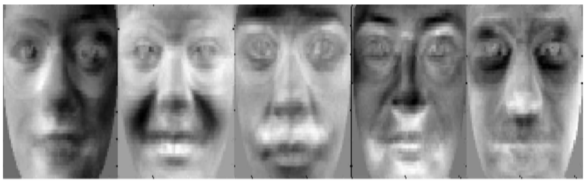


Figure 18: The first 5 Eigen faces of the whole set of faces (male and female with equal number of smile and neutral faces).

The SVM was trained in the following way:

1. Transforming the data to a format required for using the SVM software package - LIBSVM -2.83 (Chih-Chung, 2001).

2. Perform simple scaling on the data so that all the features or attributes are in the range [-1, +1].

3. Choose a kernel. We have used RBF $k(x, y) = e^{-y|x-y|^2}$ kernel.

4. Perform five fold cross validation with the specified kernel to find the best values of the parameter C and γ.

5. Use the best parameter value of C and γ to train the whole training set.

6. Finally Test.

The results of the classification are as in Table 1:

% accuracy	Test set1	Test set2
SVM on Raw faces	100	80
SVM after PCA	80	80
SVM after Gabor	95	80

Table 1: SVM Classification accuracy with faces without any pre-processing and with PCA dimensionality reduction.

It is notable and surprising that the classification using the raw images produces good generalisation on the two test sets in all cases outperforming data sets using pre-processed PCA. Some examples of misclassifications are shown in Figure 20. Whilst, some of the misclassifications are explainable some are more puzzling. The relatively poor performance of the PCA suggests that a dimensionality reduction more tuned to identifying relevant features is needed. This motivates our investigation of Gabor filter pre-processing. The images are reduced to size 64 × 64 and then Gabor processing is performed. The SVM classification results are extremely good with the fact that the images being reasonably reduced in size has not reduced the accuracy in classification. The Gabor filters have managed to pick up the relevant features from the images of half the resolution and is indicative of the power of the Gabor filters.

Figure 20: Examples of the misclassified set of faces Top row shows neutral faces wrongly classified as smiling. Bottom row shows smiling faces wrongly classified as neutral.

4 Conclusions

Identifying facial expressions is a challenging and interesting task. Our experiment shows that identification from raw images can be performed very well. However, with larger data sets, it is computationally intractable. PCA does not appear to be sufficiently tunable to identify features that are relevant for facial expression characterization. However, on performing Gabor preprocessing on the images which are reduced in size, the features are well extracted and support accurate classification.

References

Batty, B., & Taylor, M.J (2003) Early processing of the six basic facial emotional expressions, Cognitive Brain Research, 17.

Blanz, V., Sch¨olkopf, B., B¨ulthoff, H., Burges, C., Vapnik, V. & Vetter, T., "Comparison of view-based object recognition algorithms using realistic 3D models", Proc. Int. Conf. on Artificial Neural Networks 1996, 251-256.

Chih-Chung Chang & Chih-Jen Lin, LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

Cortes, C. & Vapnik, V.(1995) Support Vector Networks, Machine Learning, 20, 273-297.

Daugman, J.G, "Uncertainty relation for resolution in space, spatial frequency and orientation optimized by two dimensional visual cortical filters", Journal of Optical.Society of .America .A, Vol 2, No.7.July 1985.

Ekman, P., & Friesen, W. V. (1971) Constants across cultures in the face of the emotion, Journal of Personality and Social Psychology, 17.

Grassberger, P. & Proccacia, I. (1983). Measuring the strangeness of strange attractors. *Physica D, 9.*

Jain, A.K & Farrokhnia, F. (1991). Unsupervised texture segmentation using Gabor filters. Pattern Recognition 24(12).

Kulikowski (1982). "Theory of spatial position and spatial frequency relations in the receptive fields of simple cells in the visual cortex." Biological Cybernetics 43(3): 187-198.

Movellan, J.R. (2002). Tutorial on Gabor Filters. http://mplab.ucsd.edu/tutorials/pdfs/gabor.pdf

Philips, P.J., Grother,P., Micheals,R.J., Blackburn, D., Tabassi, E.&Bone, M.(2003). Face recognition vendor test 2002: overview and summary. Available at www.frvt.org, 2003.

Shen et.al., (2006) Review on Gabor wavelets for face recognition Pattern Anal. Applic. (2006) 9:273–292

Smith, L.I., (2002) Tutorial on Principal Component Analysis.

Zheng, D., Zhao,Y., Wang, J.,(2004a) Features Extraction Using A Gabor Filter Family, Proceedings of the sixth LASTED International conference, Signal and Image processing, Hawaii, USA.

Zheng.E., Ping, L., Song, Z.(2004b) Performance Analysis and Comparison of Neural Networks and Support Vector Machines Classifier, Proceedings of the 5th World Congress on Intelligent Control and Automation, June 15-19. 2004, Hangzhou, P.R. China.